Very Important Notes

Chapter One : Data

(1)	**Variables**	They are reserved memory locations to store values temporarily , these values can be changed during the execution of program instructions and commands.
(2)	**Constants**	They are places reserved in the (RAM) and , have data types ; this is done during the declaration. These values cannot be changed during program execution.
(3)	**Assignment Statement**	Is a statement that consists of two sides (right hand side and left hand side) separated by the assignment operator (=) ; (*which doesn't mean the arithmetic equality*). It consists of taking the value on the right side of the assignment operator (=) and storing it in the element on the left ...
(4)	**Syntax Error**	The Error that happens when writing code incorrectly.
(5)	**Logic Error**	The Error that leads to incorrect results when executing the program ; and happens if the expressions used in the assignment statement are built improperly.
(6)	**Runtime Error**	The Error that appears during the execution.

(7) &	(Concatenation Operator) : The operator that is used to join or concatenate two texts ...
(8) VbCrlf	It is a reserved word that is used to create a New Line ...
(9) " "	The two Apostrophes are used while writing or storing a text.
(10) # #	The hashes are used while writing date or time.
(11) Me	The word that means the Current Form ...
(12) REM (') Apostrophe	They are used to provides a way to add comments that help the reader understand the code written in the (Code Window) , where what is written after , is not considered lines of code and, is neglected during the program compilation using the (VB.Net) compiler ...
Specify the Scope of Declaration for Variables and Constants	
(13) Local	When declaring a Variable or Constant in an Event it can't be used out of the range of this event ...
(14) Global	When declaring a Variable or Constant on the level of classification of the (Class) . So we don't need the declaration process at each scope of the Event procedure ...
(15) Implicit Conversions	When you enter any Data on any TextBox , it considered as String by default , but (Visual Basic.Net) can convert the values to the data type that is compatible with the variable or , the property to which the value will be assigned ...

Chapter Two : Branching

(1) Conditional Expression	It is a part of a Program Code that its result can be (TRUE) or (FALSE) → asks a TRUE or FALSE Question ...
(2) If ... Then (3) If ... Then ... Else	Used only when you have only ONE Condition ... • If ... Then → when you have only ONE choice when the condition is True ... • If ... Then ... Else → when there are two alternatives ...
(4) If ... Then ... ElseIf (5) Select ... Case	Used only when you have MORE than ONE Conditional Expression ... • If ... Then ... ElseIf • Select ... Case → it is more effective when the branching depends only on the value of ONE Variable ...
(6) And - Or (Logical Operators)	Logical Operators are used to test more than One Conditional Expression together in the same IF Statement (at the same time ...)
(7) And	If both conditional expressions are (TRUE) , the result is (TRUE)
(8) Or	If either conditional expressions are (TRUE) , the result is (TRUE) → (one of them at least) ...

(9) **Isnumeric ( )**	It is one of the Predefined Functions ... It is used to check the contents of the TextBox or the Data stored in a Variable , if it is Numerical or String Data ... • Isnumeric (5) → True • Isnumeric ("Ali") → False
(10) **Mod**	It is an arithmetic operator that computes or gives the REMAINDER of the Division operation. • 8 mod 2 = 0 • 9 mod 3 = 0 • 12 mod 4 = 0 • 15 mod 5 = 0 • 18 mod 6 = 0 • 20 mod 10 = 0 • 9 mod 2 = 1 (9 – 8) • 11 mod 3 = 2 (11 – 9) • 15 mod 4 = 3 (15 – 12) • 19 mod 5 = 4 (19 – 15) • 25 mod 10 = 5 (25 – 20) • 20 mod 3 = 2 (20 – 18)
(11) **SelectedIndex**	It is a property that is used to determine the index of the item that is selected in the ListBox or ComboBox ...
(12) **Focus ()**	It is a (Method) for the TextBox and is used to set the cursor focus inside this TextBox , to begin the writing ... TextBox1 . Focus ( )

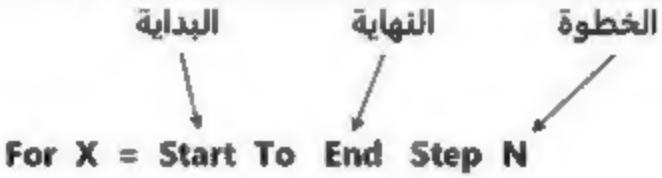
Important Comparison

Boolean (Chapter 1)	Conditional Expression (Chapter 2)
It is a Data Type , a Variable or Constant of this Type gets a value of TRUE or FALSE ... RAM X TRUE Y FALSE	It is a part of a Program Code that its result can be (TRUE) or (FALSE) ... → asks a TRUE or FALSE Question FALSE Condition ? TRUE

Chapter Three : Loops and Timers

(1) Loops	Looping is to repeat a set of statements many times ...
(2) For ... Next	It is to repeat a set of statements many times using the (For...Next) statement .
(3) Step	Using the (Step) keyword , you can increment or decrement the counter through the loop ; by the value you specify. If you do not write (Step) with the (For ...Next) statement , it means that the increment value is zero ; by default.

(4) **Next**	**The function of <u>Next</u> statement in the (For ... Next) loop :** **a) Increase the value of the counter variable with the increment value.** **b) Compare the increment value with the end value of the loop.**



- **For X = 1 To 5 → 1 , 2 , 3 , 4 , 5**
- **For X = 1 To 10 Step 2 → 1 , 3 , 5 , 7 , 9**
- **For X = 10 To 1 Step -2 → 10 , 8 , 6 , 4 , 2**
- **For X = 0.5 To 2.5 Step 0.5 → 0.5 , 1 , 1.5 , 2 , 2.5**
- **For X = 2.5 To 0.5 Step -0.5 → 2.5 , 2 , 1.5 , 1 , 0.5**
- **For X = 10 To 10 Step 2 → 10**
- **For X = 10 To 10 Step -2 → 10**
- **For X = A To B Step C → A , B , C : Variables**

(5) Do While ... Loop	**The (Do While ... Loop) is used to execute statements for an undefined number of times ; until a certain condition is met.** **This is useful if you do not know the number of iterations (repetitions) ahead.**

(6) Timers	The timer control is used to run code at regular intervals or to execute code for a specified time , it is very useful when repeating a code related to time ...
(7) Enabled	Determines if the Timer works or not ... • If Enabled is set to True → Timer is active • If Enabled is set to False → Timer is not active. And can be set programmatically through code Timer1 . Enabled = True Timer1 . Enabled = False
(8) Interval	Determines the number of milliseconds between ticks of the Timer (one second = 1000 Ticks) (one second = 1000 milliseconds)

Important Comparison

MessageBox	InputBox
A method used to display a message to the user ...  MessageBox . Show ("Hello") Or MsgBox ("Hello")	A method used to receive a value from the user and return this value in a variable of type (String)  Dim X As String X = InputBox ("enter student name")

Chapter Four : Procedures

(1) Procedure	Procedures are set of programming statements or units of code. Procedures must be called by their names , calling a procedure causes the program to execute procedure's statements or code.
(2) Sub	Sub procedures do not return a value ...
(3) Function	Functions return a value.
(4) Parameters	In the procedure declaration , we can use more than one Parameter. A Parameter allows the calling code to receive values ; that doesn't exist in the procedure and , unidentified in advance ; but specified when you call this procedure.
(5) Predefined Functions	Predefined functions are functions defined in programming languages called when a program is executed.
(6) IsNumeric ()	The function (IsNumeric) can test a value ; if it is numeric or not , where the result of this function will be (True) when the value is numeric ; and (False) when the value is non-numeric. Isnumeric (5) → True Isnumeric ("Ali") → False

(7) Show ()	The function (Show) declared within the class (MessageBox); shows a Message box . The content of this function is determined by the Parameters given, for example : • MessageBox . Show ("Hello") • Dim X As Byte = 100 MessageBox . Show (X)
(8) Now ()	The function (Now) gets The current Date and Time of the computer. MessageBox . Show (Now ( ))

Take Care ☺

Variables

Assigning values to Variables could be during the declaration or anywhere ...

```
Dim X As Byte        Or        Dim X As Byte = 5
X = 5
```

Is used on both sides of the assignment statement

```
X = Y                                Y = X
```

Constants

Is used on the right side of the assignment statement and a value is assigned to it in declaration only

```
Dim Y As Byte
Const X As Byte = 5
Y = X * 2
```

Function

Is used on the right side of the assignment statement and does not have any value.

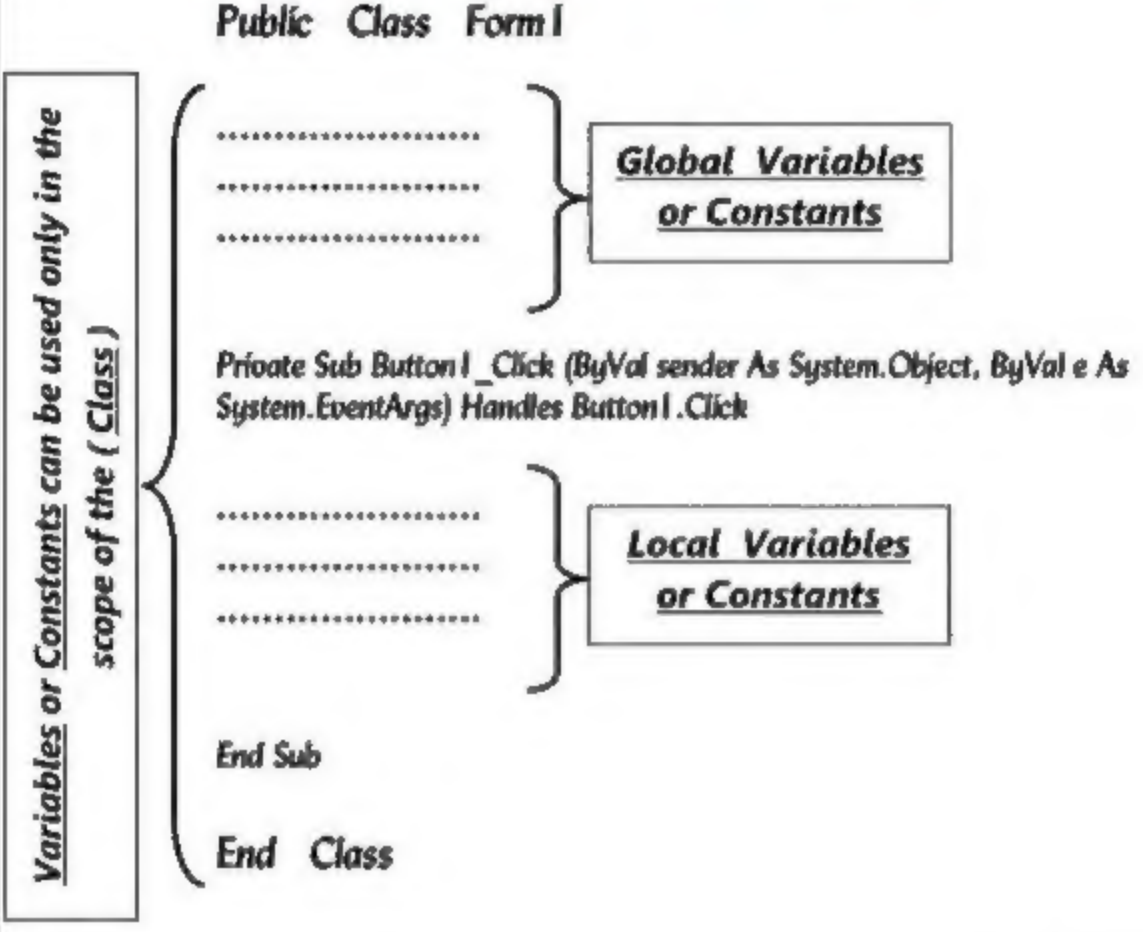
```
Dim X As Byte = TextBox1 . Text
Dim Y As Byte = TextBox2 . Text
Label1 . Text = Sum ( X , Y )
```

Sub

Is never used in the assignment statement

```
ShowOddOrEven ( N )
```

Very Important Note



- We can declare more than one Variable using the same Dim statement :

  → Have the same data type :

  Dim X , Y , Z As Byte

  → Have different data type :

  Dim X As Byte , Y As Double , Z As String

- *This (If) statement can be written, in one line without writing (End if) as follows :*

```
Private Sub Button1_Click(ByVal send

    Dim x As Single
    x = Me.TextBox1.Text
    If x >= 50 Then MsgBox("ناجح")

End Sub
```

Important Codes ☺

1) Write the code which is used to declare a variable name **"F_Name"** for storing characters :

Dim F_Name As String

2) Write the code which is used to declare a constant **"A1"** for storing the value **75.32** :

Const A1 As Single = 75.32

3) Write the code which is used to declare a constant **"D"** for storing the value **15** :

Const D As Short = 15

4) Write the code which is used to return the **current** system **Date** and **Time** in **Label1** is :

Label1 . Text = Now ( )

5) Write the code which is used to **activate** the **Timer1** Control :

Timer1 . Enabled = True

6) Write the code which is used to deactivate or **stop** the **Timer1** Control from **working** :

Timer1 . Enabled = False

7) Write the code which is used to set the time interval of **Timer1** in **2 seconds** :

Timer1 . Interval = 2000

8) Write the For statement to show the **Odd** numbers from **11** to **99** :

For X = 11 To 99 Step 2

9) Write the For statement to show the **Even** numbers from **11** to **99** :

For X = 12 To 99 Step 2

10) Write the code to print out the variable (**total**) on **Label3** :

Label1 . Text = total

11) Write the code that declare a place for the value **pi** with a suitable data type , where **pi = 3.14** :

Const pi As Single = 3.14

12) Write the code that **set** the **cursor** inside **TextBox1** :

TextBox1 . Focus ( )

13) Write the code that **empty TextBox1** :

TextBox1 . Text = " "